

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-007138

(43)Date of publication of application : 11.01.2002

(51)Int.Cl. G06F 9/445

G06F 3/00

G06F 13/00

(21)Application number : 2001-107647

(71)Applicant : FUJITSU LTD

(22)Date of filing :

05.04.2001

(72)Inventor : MASUDA TAKAHIRO
TOGAWA YOSHIFUSA

(30)Priority

Priority number : 2000115239 Priority date : 17.04.2000 Priority country : JP

(54) METHOD FOR INSTALLING AND STARTING APPLICATION, DEVICE FOR EXECUTING THE SAME, AND MEDIUM AND PROGRAM

(57)Abstract:

PROBLEM TO BE SOLVED: To automatically install an application program whose version is optimal to a data file without making a user to perform any troublesome version judgment, and to automatically start the application.

SOLUTION: A document file to be read is analyzed so that a feature point such as the reserved word of a macro-instruction can be extracted, and the version of an application suitable for the document file is identified according to the extracted feature point. Then, whether an application program of the identified version is already installed is judged, and when it is judged that such application program has not been installed, the application program of the pertinent version is newly installed.

LEGAL STATUS

[Date of request for examination] 23.04.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against
examiner's decision of rejection]

[Date of requesting appeal
against examiner's decision of
rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-7138

(P2002-7138A)

(43) 公開日 平成14年1月11日 (2002.1.11)

| (51) Int.Cl. ⁷ | 識別記号 | F I | テーマコード [*] (参考) |
|---------------------------|-------|--------------|--------------------------|
| G 0 6 F 9/445 | | G 0 6 F 3/00 | 6 5 1 A 5 B 0 7 6 |
| 3/00 | 6 5 1 | 13/00 | 5 3 0 B 5 E 5 0 1 |
| 13/00 | 5 3 0 | 9/06 | 6 1 0 L |

審査請求 未請求 請求項の数14 O L (全 17 頁)

(21) 出願番号 特願2001-107647 (P2001-107647)
(22) 出願日 平成13年4月5日 (2001.4.5)
(31) 優先権主張番号 特願2000-115239 (P2000-115239)
(32) 優先日 平成12年4月17日 (2000.4.17)
(33) 優先権主張国 日本 (J P)

(71) 出願人 000005223
富士通株式会社
神奈川県川崎市中原区上小田中4丁目1番1号
(72) 発明者 増田 高弘
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
(72) 発明者 外川 好房
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
(74) 代理人 100089244
弁理士 遠山 勉 (外1名)

最終頁に続く

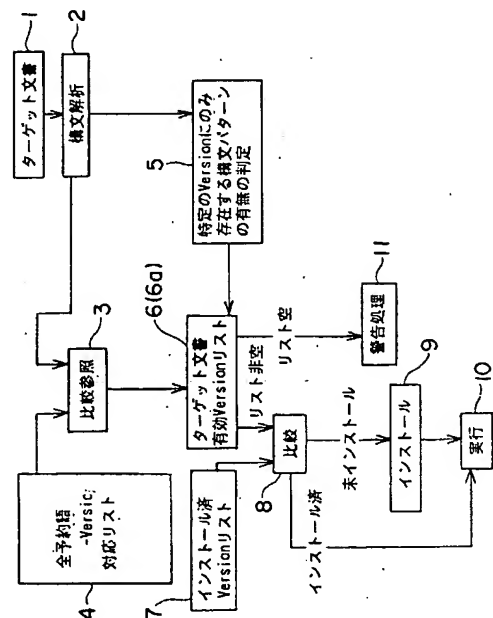
(54) 【発明の名称】 アプリケーションのインストール方法、起動方法、実行装置、媒体およびプログラム

(57) 【要約】

【課題】 ユーザーにわずらわしいバージョンの判定を行わせることなく、データファイルに対して最適なバージョンのアプリケーションプログラムを自動インストールし、当該アプリケーションの自動起動を可能にする。

【解決手段】 読み出そうとする文書ファイルを解析してマクロ命令の予約語等のような特徴点を抽出し、抽出された特徴点によってその文書ファイルに適したアプリケーションのバージョンを識別する。そして、前記で識別されたバージョンのアプリケーションプログラムが既にインストールされているか否かを判定し、この判定結果によりインストールされていないときには当該バージョンのアプリケーションプログラムを新たにインストールするようにした。

本発明の実施例による自動インストールシステムの機能ブロック図



【特許請求の範囲】

【請求項1】 データファイルを解析して特徴点を抽出するステップと、

抽出された特徴点に基づいて当該データファイルに適したアプリケーションのバージョンを決定するステップと、

決定されたバージョンのアプリケーションプログラムが既にインストールされているか否かを判定するステップと、

インストールされていないときに当該バージョンのアプリケーションプログラムを新たにインストールするステップとからなるアプリケーションのインストール方法。

【請求項2】 前記特徴点はデータファイルに含まれるマクロ命令の予約語であり、バージョン毎に特有な予約語を検出することにより前記バージョンを決定する請求項1記載のアプリケーションのインストール方法。

【請求項3】 前記特徴点はデータファイルに含まれるマクロ命令の構文パターンであり、バージョン毎に特有な構文パターンを検出することにより前記バージョンを決定する請求項1記載のアプリケーションのインストール方法。

【請求項4】 2以上のデータファイルを解析してそれぞれの特徴点を抽出するステップと、

前記特徴点に基づいて前記データファイルを可読可能なアプリケーションプログラムのバージョンを決定するステップと、

前記データファイルと、前記で決定されたバージョンのアプリケーションプログラムとを同時に表示するステップと、

表示されたアプリケーションプログラムを前記いずれかのデータファイルと関連付けて起動するステップとからなるアプリケーションの起動方法。

【請求項5】 前記データファイルは、前記アプリケーションプログラムの対応バージョン毎に異なるシンボル図形で表示される請求項4記載のアプリケーションの起動方法。

【請求項6】 前記アプリケーションプログラムが選択されたときに、当該アプリケーションプログラムのバージョンに対応したデータファイルのみが表示される請求項4記載のアプリケーションの起動方法。

【請求項7】 前記データファイルに対して対応するアプリケーションプログラムが存在しないときに、対応するアプリケーションプログラムのインストールを実行する請求項4記載のアプリケーションの起動方法。

【請求項8】 前記アプリケーションプログラムのインストールを実行する際に空き領域が不足している場合に既存のファイルを削除するステップを含む請求項4記載のアプリケーションの起動方法。

【請求項9】 2以上のデータファイルを解析してそれぞれの特徴点を抽出するステップと、

前記特徴点に基づいて前記データファイルを可読可能なアプリケーションプログラムのバージョンを決定するステップと、

前記データファイルと、前記で決定されたバージョンのアプリケーションプログラムとを同時に表示するステップと、

表示されたアプリケーションプログラムを前記いずれかのデータファイルと関連付けて起動するステップとを含むプログラムを記録した記録媒体。

10 【請求項10】 2以上のデータファイルを解析してそれぞれの特徴点を抽出する手段と、

前記特徴点に基づいて前記データファイルを可読可能なアプリケーションプログラムのバージョンを決定する手段と、

前記データファイルと、前記で決定されたバージョンのアプリケーションプログラムとを同時に表示する手段と、

表示されたアプリケーションプログラムを前記いずれかのデータファイルと関連付けて起動する手段とからなるアプリケーションプログラム実行装置。

20 【請求項11】 データファイルを解析して特徴点を抽出する手段と、

抽出された特徴点に基づいて当該データファイルに適したアプリケーションのバージョンを決定する手段と、

決定されたバージョンのアプリケーションプログラムが既にインストールされているか否かを判定する手段と、インストールされていないときに当該バージョンのアプリケーションプログラムを新たにインストールする手段とからなるアプリケーションインストール装置。

30 【請求項12】 データファイルを解析して特徴点を抽出するステップと、

抽出された特徴点に基づいて当該データファイルに適したアプリケーションのバージョンを決定するステップと、

決定されたバージョンのアプリケーションプログラムが既にインストールされているか否かを判定するステップと、

インストールされていないときに当該バージョンのアプリケーションプログラムを新たにインストールするステップとを順次実行するプログラムを記憶した記憶媒体。

40 【請求項13】 2以上のデータファイルを解析してそれぞれの特徴点を抽出するステップと、

前記特徴点に基づいて前記データファイルを可読可能なアプリケーションプログラムのバージョンを決定するステップと、

前記データファイルと、前記で決定されたバージョンのアプリケーションプログラムとを同時に表示するステップと、

50 表示されたアプリケーションプログラムを前記いずれかのデータファイルと関連付けて起動するステップとを含む

むプログラム。

【請求項14】 データファイルを解析して特徴点を抽出するステップと、
抽出された特徴点に基づいて当該データファイルに適したアプリケーションのバージョンを決定するステップと、
決定されたバージョンのアプリケーションプログラムが既にインストールされているか否かを判定するステップと、
インストールされていないときに当該バージョンのアプリケーションプログラムを新たにインストールするステップとを順次実行するプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明はデータに必要なバージョンのアプリケーションの自動インストールおよび自動起動の技術に関する。

【0002】

【従来の技術】 パーソナルコンピュータやワードプロセッサ専用機で作成された文書ファイルやスプレッドシートファイルのデータは、これを作成・更新するアプリケーションプログラムのバージョンアップにしたがってデータ形式が変更されるため、新しいバージョンのプログラムで作成されたデータが旧バージョンのプログラムでは利用できないことが多い。逆に新しいバージョンのアプリケーションを使っている場合でも、旧バージョンのアプリケーションで作成されたデータが編集できないことがある。

【0003】 ワードプロセッサのアプリケーションの場合、アプリケーション側のバージョンによってはデータ格納形式が異なるため、文書のレイアウトが別のバージョンのアプリケーションプログラムでは再現できない場合がある。そのために、文書コンバータと呼ばれる別のアプリケーションを介在させてデータ形式を変換させる必要があった。

【0004】 また、スプレッドシートと呼ばれる表計算プログラムの場合、バージョンによって使用可能なオブジェクトが異なるため、異なるバージョンで作成されたデータのあるプログラムで読み出そうとした場合、エラーとなることがある。

【0005】 さらに、この種のプログラムではマクロと呼ばれる一定の手順を記述した簡易プログラムが付属しているが、このマクロの仕様、たとえば予約語の有無などもプログラムのバージョンによって異なるため、作成したバージョンと異なるバージョンのプログラムでデータファイルを開いた場合に当該マクロが実行不可能となりエラーを生じる可能性があった。

【0006】 インターネットで代表されるネットワーク社会では、通信を経由してこの種のデータファイルがやりとりされることが多く、受信したデータファイルが、

自分のコンピュータにインストールされているアプリケーションプログラムで利用できるバージョンであるか否かは重要な問題だった。

【0007】 したがって、あるユーザーがインストールされているアプリケーションのバージョンでそのデータファイルが扱えない場合には、ユーザーは利用可能なアプリケーションのバージョンを判定して、そのアプリケーションの入手先を調査し、入手後に自分のコンピュータ上にインストールする必要があった。

【0008】

【発明が解決しようとする課題】 ところで、バージョンによって異なる拡張子が付与されるデータファイルであれば拡張子を参照するだけで利用可能なアプリケーションプログラムのバージョンを知ることができる。

【0009】 しかし、docなどの文書形式のようにバージョンが異なっても統一した拡張子が付与されるアプリケーションプログラムの場合には、データファイルを外部から観察しただけでは適切なバージョンを判定することは困難だった。そのため、最悪の場合にはユーザーは種々のバージョンのアプリケーションプログラムを試行的にインストールしてみてデータファイルのオープンを繰り返してみる必要があった。

【0010】 本発明は、このような点に鑑みてなされたものであり、ユーザーにわずらわしいバージョンの判定を行わせることなく、データファイルに対して最適なバージョンのアプリケーションプログラムを自動インストールし、当該アプリケーションの自動起動を可能にすることを技術的課題とする。

【0011】

【課題を解決するための手段】 本発明は、まず読み出そうとするデータファイルを解析して特徴点を抽出し、抽出された特徴点によってそのデータファイルに適したアプリケーションのバージョンを識別する。そして、前記で識別されたバージョンのアプリケーションプログラムが既にインストールされているか否かを判定し、この判定結果によりインストールされていないときには当該バージョンのアプリケーションプログラムを新たにインストールするようにした。

【0012】 この特徴点とは、たとえば文書ファイルにおけるマクロ命令の予約語や構文パターンである。これらの予約語や構文パターンはそれを作成したアプリケーションプログラムのバージョンに依存していることが多く、これらを抽出することによって当該文書ファイルを開くための最適なアプリケーションプログラムのバージョンを知ることができる。

【0013】 また、バージョンの解析結果に基づいて文書ファイルと対応するバージョンのアプリケーションプログラムとを同時に表示装置上に表示し、いずれのバージョンのアプリケーションプログラムで文書ファイルを開くかをユーザーに選択させるようにしてもよい。この

場合、解析結果で得られた情報に基づいて文書ファイルのバージョン毎に異なるアイコンを付記してもよいし、アプリケーションプログラムをまず選択させた場合、このアプリケーションプログラムで開くことのできる文書ファイルだけを表示させるようにしてもよい。

【0014】

【発明の実施の形態】

【0015】

【実施例1】以下、図面に基いて、本発明の実施の形態を説明する。図1は、本発明の実施例による自動インストールシステムの機能ブロック図、図2は本実施例のシステム構成を示すブロック図である。

【0016】図2に示すように、ネットワーク30を介してサーバ31と端末装置21とが接続されている。端末装置21はそれ自体でコンピュータシステムを構成しており、本実施例の自動インストール処理は端末装置21だけでも実施可能である。

【0017】端末装置21は、バス22を介して中央処理装置(CPU)23、メモリ24、表示装置(CRT)25、印刷装置(PRT)26およびハードディスク装置(HD)27を有している。また、外部補助入力手段としてマウス29を有している。また、前記バス22は、通信インターフェース28を介して前記ネットワーク30と接続されている。

【0018】図1は、本実施例の自動インストール手順を示しており、かかる機能はハードディスク装置(HD)27に格納されたプログラムがメモリ24に読み込まれて、さらに該プログラムを中央処理装置(CPU)23が順次実行することにより実現される。

【0019】ターゲット文書1は、たとえばワードプロセッサ用の文書ファイルやスプレッドシート用のデータファイルであり、ハードディスク(HD)27内に登録されている。ターゲット文書1は、通常の場合、文書領域とマクロ記述領域とを有しており、所定のワードプロセッサプログラムで当該ターゲット文書1を読み込んだ場合に、その文書に付随したマクロ命令で規定された文字列の処理や整形が可能となっている。

【0020】このようなターゲット文書1がメモリ24に読み込まれると、中央処理装置(CPU)23はこのターゲット文書1に対して構文解析処理2を行う。この構文解析処理は、具体的には、マクロ命令予約語を抽出する作業である。

【0021】マクロ命令予約語の抽出は、ターゲット文書1から予約語リスト12を生成することにより行われる。これは具体的には、ハードディスク装置(HD)27に登録された全予約語データベース(DB)13を参照しパターンマッチングによりターゲット文書1の中から予約語を抽出し、これを予約語リスト12に登録することである。そして、この予約語リスト12とハードディスク装置(HD)27内に登録された全予約一バジ

ョン対応リスト4とが比較されて(比較参照処理3)、当該ターゲット文書1に対する有効バージョンリスト6が生成される。

【0022】また、前述の構文解析処理2と並行して、中央処理装置(CPU)23は構文パターンの検出を行う(ステップ5)。この構文パターンの検出では、ターゲット文書1中のマクロプログラムをトークンに分割してそらのトークンの属性を判定した上で特定の属性のトークンの結合状態が存在するか否かを判定することにより実現する。たとえば、「Collection-Classのインスタンスとなる変数」+「(」+「Strings-Classのインスタンスとなる変数」+「)」というような結合状態を検出することである。有効バージョンリスト6aの生成に際しては、この構文パターンの検出結果も反映させて最終的な有効バージョンリスト14を生成する。このとき、バージョンリスト6aの生成に際しては、マクロ命令予約語の抽出結果と構文パターンの検出結果とのいずれを優先させてもよい。たとえば、マクロ命令予約語の抽出結果に基づくバージョンと、構文パターンの検出結果に基づくバージョンとが異なるときには、より数値の大きい(最新のバージョン)を優先的に採用して有効バージョンリスト6aを生成してもよい。

【0023】次に、前記有効バージョンリスト6aと、インストール済みバージョンリスト7とが比較される(比較処理8)。この比較処理8の結果、該当するバージョンのアプリケーションプログラムが既に端末装置21内のハードディスク装置(HD)27内に実行可能な状態でインストールされている場合には当該アプリケーションプログラムを実行し(実行処理10)、インストールされていない場合には、インストール処理9を行う。

【0024】なお、有効バージョンリスト6(6a)の生成の際にバージョンリスト内が空である場合、すなわち有効バージョンが判定できない場合には、警告処理11がなされる。この警告処理11は、たとえば表示装置(CRT)25上に視覚的な警告表示を行うことが考えられる。

【0025】

【実施例2】実施例1では、比較処理8において、中央処理装置(CPU)23が比較処理8を実行し、最適なアプリケーションプログラムのバージョンを自動決定する方式を説明したが、本実施例2では、比較処理8の結果を表示装置(CRT)25上でユーザーが一見して把握可能な形式で表示し、ユーザー自身に複数のバージョンのアプリケーションプログラムを選択させる余地を有している点が特徴である。

【0026】なお、図1で説明した処理の中で、比較処理8以前の処理については実施例1と同様であるので説明を省略する。中央処理装置(CPU)23は、ターゲット文書1から得られた有効バージョンリスト6(6

a)とインストール済みバージョンリスト7とを比較する際に、表示装置(CRT)25上に図4に示すような比較画面41を表示する。この比較画面41は、同図左側のアプリケーションリスト表示窓42と、モジュールリスト表示窓43との分割されている。アプリケーションリスト表示窓42は、アプリケーションプログラムがグループ毎に階層的に表示可能となっており、同図において、スプレッドシート(表計算プログラム)のフォルダ内にバージョン1.0～3.1がアプリケーションプログラムとして列記されている。

【0027】また、閉じた状態とはなっているが、データベースプログラムとワードプロセッサプログラムも同様にフォルダ内に複数のアプリケーションプログラムを有している。

【0028】一方、モジュールリスト表示窓43には、モジュールとしてのターゲット文書44(図1ではターゲット文書1)が複数表示されるようになっている。ここで、ユーザーは表示装置(CRT)を参照して、どの文書(モジュール)をどのバージョンのアプリケーションプログラムで開くかを任意に決定することができる。

【0029】同図において、たとえばターゲット文書44(モジュール)をスプレッドシートのバージョン2.0のアプリケーションプログラムで開きたい場合、マウス23によって画面上のマウスカーソル46に配置し、ここで一旦マウス29のボタン(図示せず)を押す。この行為(ドラッグ)によって、ターゲット文書44(モジュール)が選択されたことになり、アプリケーションリスト表示窓42では、このターゲット文書44を開くことのできるバージョンのアプリケーションプログラムのみが高い輝度表示で表示される。すなわち、当該ターゲット文書44を開くことのできるアプリケーションプログラムのバージョンが複数ある場合には、該当する複数のバージョンの全てが高輝度で表示される。

【0030】次に、ユーザーは、前記マウス29のボタンを押した状態(ドラッグ状態)のままマウスカーソル46をアプリケーション表示窓42の任意のバージョン(ここではバージョン2.0)のアプリケーションプログラム上に移動させて、前記ボタンを開放する(ドラッグ)ことによってスプレッドシートのバージョン2.0のアプリケーションプログラムがターゲット文書44(モジュール)を対象ファイルとして起動される。この手法は、いわゆる「ドラッグ・アンド・ドロップ」と呼ばれる画面インターフェース上の入力手法である。

【0031】図4で説明したインターフェースを実現するための機能構成を示したものが図5である。同図において、文書解析制御部51は、図1における構文解析処理2および比較参照処理3を行うとともに、表示装置

(CRT)25への表示制御を行う機能を有している。すなわち、文書解析制御部51は、所定のフォルダまたはディレクトリ配下にあるターゲット文書1を順次読み

込んでそのバージョンの解析を行い、モジュールリスト表示窓43にリスト形式で表示する。このモジュールリスト表示窓43に表示されたモジュール(ターゲット文書44)は、ハードディスク装置(HD)27内にインストールされたアプリケーションプログラムで読み込むことのできる正当な文書ファイルであるか否か、正当な文書ファイルである場合にはどのバージョンで開くことが可能なのか否かが文書解析制御部51によって既に識別された状態となっている。

10 【0032】文書解析制御部51は、ターゲット文書1を読み込んで解析を行った際に、この解析結果であるアプリケーション名とバージョン名とをアプリケーションパス取得部54に通知する。このアプリケーションパス取得部54はアプリケーションリスト表示窓42の表示を制御しており、文書解析制御部51から受け取ったバージョンのアプリケーションのみをアプリケーションリスト表示窓54に表示させる。

20 【0033】また、モジュールリスト表示窓43のターゲット文書44(モジュール)がマウス29のドラッグ操作で前述のように選択された際には、この選択情報が文書解析制御部51を通じてアプリケーションパス取得部54に通知され、アプリケーションリスト表示窓42中の対応するアプリケーションプログラムのバージョンが高輝度表示されるようになっている。

30 【0034】そして、マウス29のドロップ操作で所定のバージョンのアプリケーション(ここではバージョン2.0)のスプレッドシートのアプリケーションプログラム45)が選択されると、この選択情報がアプリケーションパス取得部54を通じて実行パスとしてOS(Operating System)に通知される。

【0035】これによりOSは、アプリケーションパス取得部54より実行ファイルパスを、文書解析制御部51より文書ファイルパスを受け取り、対象となるターゲット文書44(モジュール)を選択されたバージョンのアプリケーションプログラムで実行することになる。

40 【0036】なお、本実施例においてOSとしては、マイクロソフト社のウィンドウズ3.1、95または98等を用いることができる。図6および図7は比較画面41の表示において、ターゲット文書(モジュール)と対応するアプリケーションプログラムのバージョンとの対比表示の変形例である。

【0037】図6は、アプリケーションプログラムのバージョン毎に異なるアイコンを定義し、ターゲット文書44(モジュール)を表示する場合に対応するバージョンのアイコンで表示させるようにしたものである。

50 【0038】この点について、既存のOSでは、ファイル名の拡張子(たとえばXLSやDOC)によってアイコンを定義しているため、同一の拡張子が付与されるとバージョンが異なっても同じアイコンで表示されてしまうため、アイコンを目視しただけでは当該ター

ゲット文書のバージョンの把握が困難だった。

【0039】この点について図6に示した例では、文書解析制御部51において全てのターゲット文書44についてその対応バージョンが把握されているため、ファイル名の拡張子に左右されることなくターゲット文書の対応バージョンがアイコンによって一目で識別可能となるように表示できる。

【0040】図7は、アプリケーションリスト表示窓42で指定したバージョンに対応したターゲット文書44のみをモジュールリスト表示窓43に表示するようにした例である。この場合には、アプリケーションリスト53で特定のバージョンが指定されると、アプリケーションパス取得部54がこれを検出し、文書解析制御部51に当該バージョンが指定されていることを通知する。この通知に基づいて文書解析制御部51では前記バージョンに対応したターゲット文書（モジュール44）のみをモジュールリスト表示窓43に表示させることができる。

【0041】図8は、ターゲット文書によってアプリケーションプログラム53のバージョンを選択する概念を示している。図4、図6、図7で説明した画面インターフェースでターゲット文書84a、84bが文書解析制御部51に読み込まれると、これが解析詳細部51aで解析されてインストール済みのアプリケーションプログラムの中から最適なバージョンのアプリケーションプログラム45aを選択するようになっている（図8の上図）。そして、解析詳細部51aで決定されたアプリケーションプログラム45aによってターゲット文書84aが開かれるようになっている。

【0042】図9は、文書解析制御部51による解析の結果、アプリケーションリスト53内に該当するバージョンのアプリケーションプログラムが存在しない場合の機能ブロック構成を示している。同図において大半の構成は図5と同様であるのでその部分の説明は省略する。

【0043】インストール制御部91は、ハードディスク装置（HD）27内にアプリケーションプログラムを実行可能な状態にインストールするための制御部であり、具体的には端末装置21の図示しないCD-ROMやサーバ31からアプリケーションプログラムをハードディスク装置（HD）27内にインストールする機能を有している。すなわち、図9では、アプリケーションリスト53からアプリケーションパス取得部54がパスを取得できない場合、つまり、モジュールリスト52で選択されたターゲット文書44（モジュール）に対応するバージョンが存在しない場合に、インストール制御部91が起動されるようになっている。

【0044】図10は、インストール制御部91による新たなバージョンのアプリケーションプログラムのインストール手順を示したものである。同図では、バージョンnのターゲット文書1001、1002が順次文書解

析制御部51に読み込まれ、解析詳細部51aでアプリケーションプログラムのバージョンを判定したにもかかわらず、インストール済みのアプリケーションプログラム（1003）はバージョンmであり、前記バージョンnのターゲット文書1001、1002に対応していない状態となっている（図10（a））。

【0045】このとき、文書解析制御部51からの通知を受けて、インストール制御部91が起動し、ターゲット文書1001、1002のバージョンnに対応したアプリケーションプログラム1004が新たにインストールされる。（図10（b））。

【0046】ここで、当該アプリケーションプログラムのインストール元は、端末装置21内のハードディスク装置（HD）27であってもよいし、図示しないCD-ROMドライブ装置からであってもよいし、さらに、ネットワーク30を介したサーバ31であってもよい。ネットワーク30を介してサーバ31からアプリケーションプログラムをダウンロードする場合には、TCP/IPによるFTP（File Transfer Protocol）を用いることができる。

【0047】ハードディスク装置（HD）27内に新たにインストールされたアプリケーションプログラム1004が起動して、これに対応したバージョンのターゲット文書1001または1002が開かれる（図10（c））。

【0048】図11および図12は、必要なバージョンのアプリケーションプログラムをインストールする際に、インストール領域を確保するための機能ブロック図、その処理手順を示す図である。

【0049】図11の構成は図9で示した構成にアンインストール制御部1101が付加されている点特徴である。すなわち、図9および図10で説明した例でインストール制御部91が起動されたにもかかわらず、ハードディスク装置（HD）27内に新たなバージョンのアプリケーションプログラムをインストールする領域が不足している場合がある。

【0050】この場合には、図11で示すアンインストール制御部1101が起動される。このアンインストール制御部1101もプログラムとして実現されており、中央処理装置（CPU）23が当該プログラムを実行することにより具体化される。

【0051】図12は、上記アンインストールおよびインストールの処理手順を示している。解析詳細部51aの解析によってインストール済みのアプリケーションプログラムから最適なバージョンのものが検出できなかった場合（図12（a））、インストール制御部91が起動するが、ハードディスク装置（HD）27に新たなアプリケーションプログラムをインストールする領域が残っていない場合がある（図12（b））。

【0052】このような場合に、インストール制御部9

1はアンインストール制御部1101に対して領域不足を通知する。これによって、アンインストール制御部91はハードディスク装置(HD)27中から不要なファイルを削除する。このときにいずれのファイルを削除するか否かについては、下記のような基準で決定することができる。

【0053】たとえば、利用条件を満たしていないアプリケーションプログラムを優先的にアンインストールするようにしてもよい。たとえば、ダウンロード後30日間のみ試用可能な条件が付加されているようなアプリケーションプログラムがこれに該当する。

【0054】また、容量順にアンインストールすることも可能である。一般に、容量が少ないアプリケーションほど重要度が低い傾向がある場合、容量が小さなアプリケーションプログラムから先に削除する。また、アンインストールするプログラム本数を少なくして領域確保効率を優先させるのであれば、容量が大きなアプリケーションプログラムから削除する。

【0055】また、インストールした日付順、重要度、作成文書数、共通ファイルの数の大小等によってアンインストールする順番を決定してもよい。図13および図14は、ネットワーク30を通じて端末装置21がサーバ31からアプリケーションプログラムをダウンロードする手順を示すシーケンス図である。

【0056】同図においてまず、サーバ31は管理プログラムを、端末装置21はクライアント用プログラムをそれぞれ起動する。なお、端末装置21のハードディスク装置(HD)27内には、端末装置21がサーバ31から取得したファイルの履歴がファイル取得履歴ファイル1301として格納されている。

【0057】端末装置21のインストール制御部91によりインストール処理が開始されると、まず端末装置21よりサーバ31に対して利用開始メッセージが送信される。これにより、サーバ31がインストール開始を認識すると、自身の自動管理システムが有効であることを端末装置21に通知する。

【0058】次に、端末装置21は、利用者識別符号(ID)をサーバ31に通知する。これを受領したサーバ31は、端末装置21に対してハードウェア情報を要求する。このハードウェア情報とは、端末装置21の中央処理装置(CPU)23の種別、クロック周波数(Clock)、OS、メモリ容量、ハードディスク装置(HD)27の容量等である。これらの情報を受け取ると、次に端末装置21に対してファイル取得履歴を要求する。このファイル取得履歴とは、いわゆる履歴であり、ファイル名、URL(Uniform Resource Locator)、圧縮方式、バージョン等の情報であり、前述のファイル取得履歴ファイル1301として端末装置21がハードディスク装置(HD)27が有しているものである。

【0059】サーバ31がこのファイル取得履歴を受領

すると、サーバはこれに基づいて過去にダウンロードしたファイルの新しいバージョンが既にサーバにダウンロード可能な状態で格納されているか否かを判定する。そして、新しいバージョンが存在しない場合には、端末装置21にはサーバ31からみて最新のバージョンのアプリケーションプログラムが既にインストール済みであるため、処理を終了する。

【0060】一方、端末装置21から得たファイル取得履歴よりもさらに新しいバージョンがサーバ31側に存在する場合には、必要なインストーラ(インストールを行うための補助プログラム)を選択し、これを端末装置21に転送する。

【0061】端末装置21からサーバ31に前記インストーラの転送終了が通知されると、サーバ31は端末装置21に対してファイル取得履歴の書き換えを命令する。端末装置21はこの命令に基づいて、サーバ31にバージョンやファイル日付等の必要項目を要求し、サーバ31はこれに応答して前記項目の情報を端末装置21に通知する。

【0062】端末装置21では、前記項目情報に基づいて自身のファイル取得履歴ファイル1301を書き換えるとともに、サーバ31とのネットワークを切断する。そして、端末装置21では、ダウンロードしたインストーラを起動してインストールを開始する。ここでインストールが正常終了しなかった場合には、アーカイブの正当性検査を開始する。このアーカイブ正当性検査は、再度サーバ31とのネットワーク接続を行い、サーバ31に対して検査要求を送信することによって行われる。この検査結果に基づいて必要な場合には前記インストーラを再度サーバ31から端末装置21に転送する。

【0063】

【発明の効果】本発明によれば、ファイルの外観からは特定できないデータファイルに対して、最適なバージョンのアプリケーションプログラムを決定することができる。とともに、当該アプリケーションプログラムを自動インストールし、当該アプリケーションの自動起動を行うことができる。

【図面の簡単な説明】

【図1】 本発明の実施例による自動インストールシステムの機能ブロック図

【図2】 実施例1のシステム構成を示すブロック図

【図3】 実施例1の各リストの関係を示すブロック図

【図4】 実施例2の表示装置における画面表示例

【図5】 図4の画面構成を実現するための機能ブロック図

【図6】 実施例2の表示装置における画面表示例(2)

【図7】 実施例2の表示装置における画面表示例(3)

【図8】 実施例2の文書ファイルとアプリケーション

1 ターゲット文書

4 全予約語バージョン対応リスト
6 ターゲット文書有効バージョンリスト
7 インストール済みバージョンリスト

| | |
|-----|----------------|
| 2 1 | 端末装置 |
| 2 2 | バス |
| 2 3 | 中央処理装置 (C P U) |
| 2 4 | メモリ |

25 表示装置 (C R T)
26 印刷装置 (P R T)
27 ハードディスク装置 (H D)

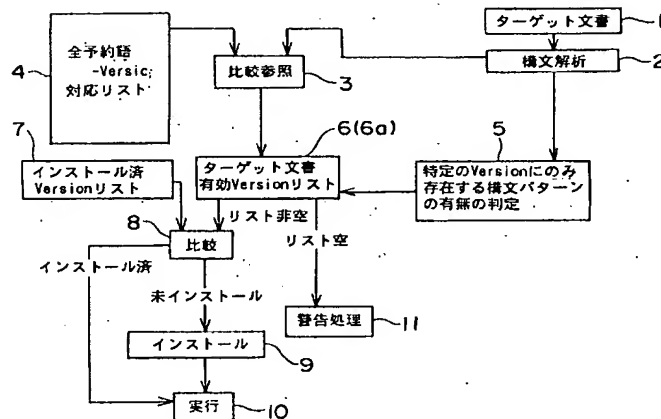
28 通信インターフェース
29 マウス

30 ネットワーク
31 サーバ

4 1 比較画面
4 2 アプリケーションリスト表示窓
4 3 モジュールリスト表示窓
4 6 マウスカーソル

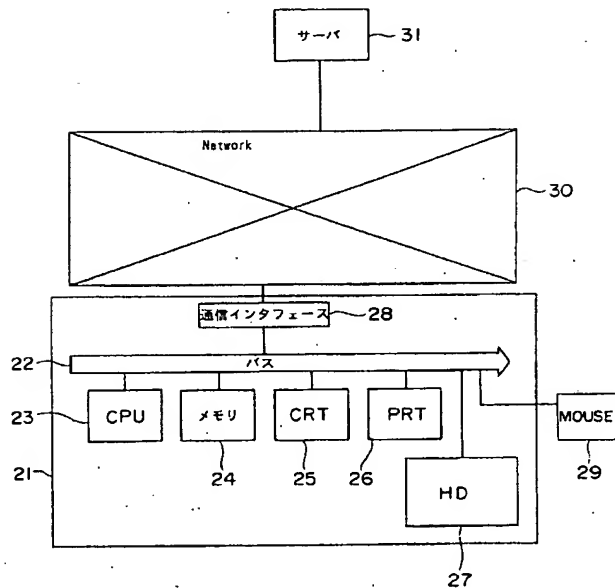
20

本発明の実施例による自動インストールシステムの機能ブロック図



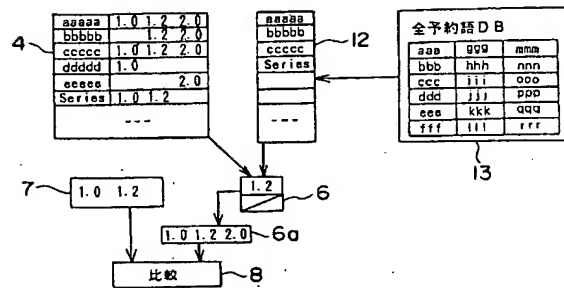
【図2】

実施例1のシステム構成を示すブロック図



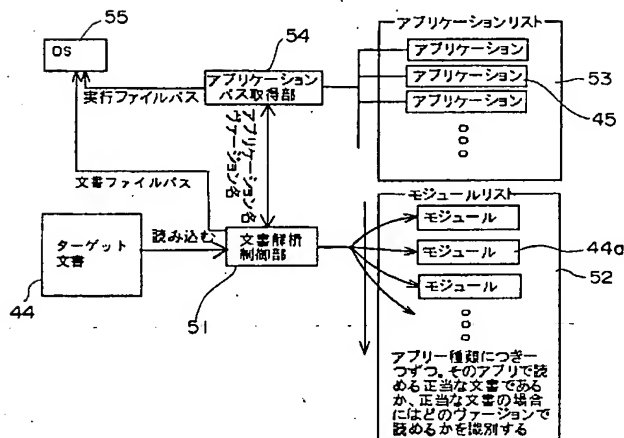
【図3】

実施例1の各リストの関係を示すブロック図



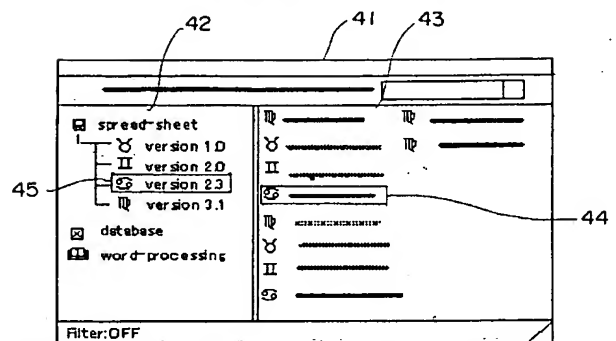
【図5】

図4の画面構成を実現するための機能ブロック図



【図6】

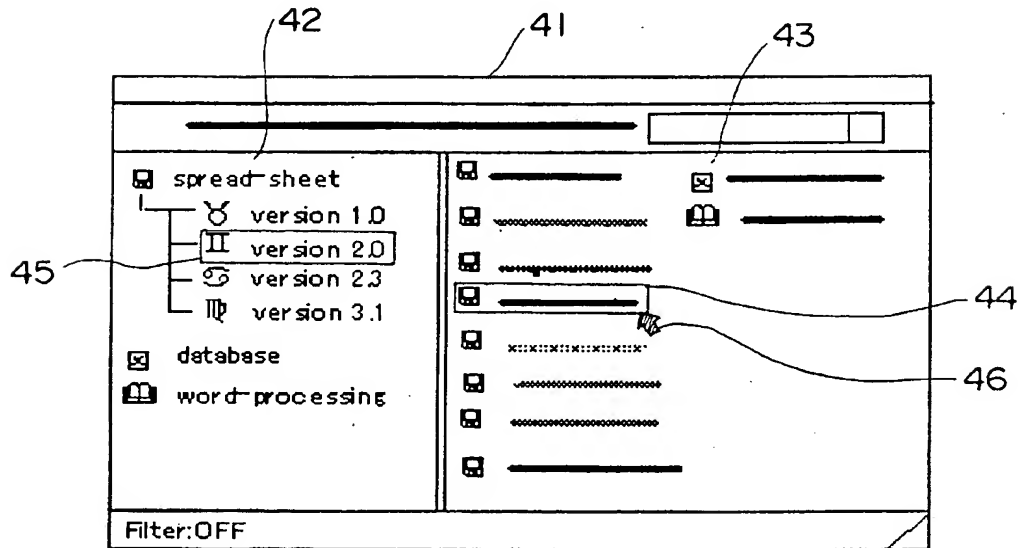
実施例2の表示装置における画面表示例(2)



通常同一のアプリケーションにより作成された文書はバージョンによらず同一のアイコンを伴うことが多いが、たとえば図に示すようにバージョンによりその文書アイコンを視覚的に変えて表示することにより、利用者に対する利便がはかられる。

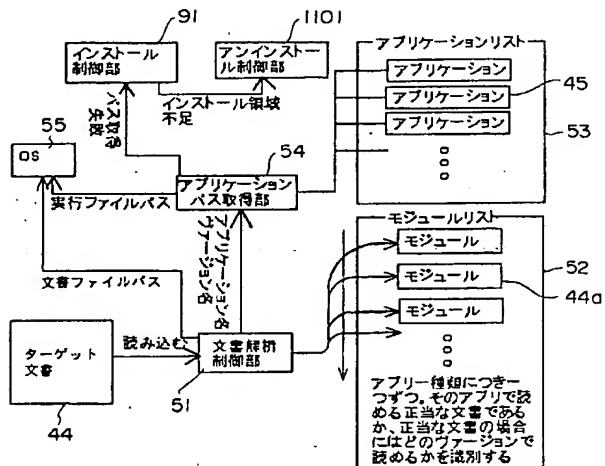
【図 4】

実施例 2 の表示装置における画面表示例



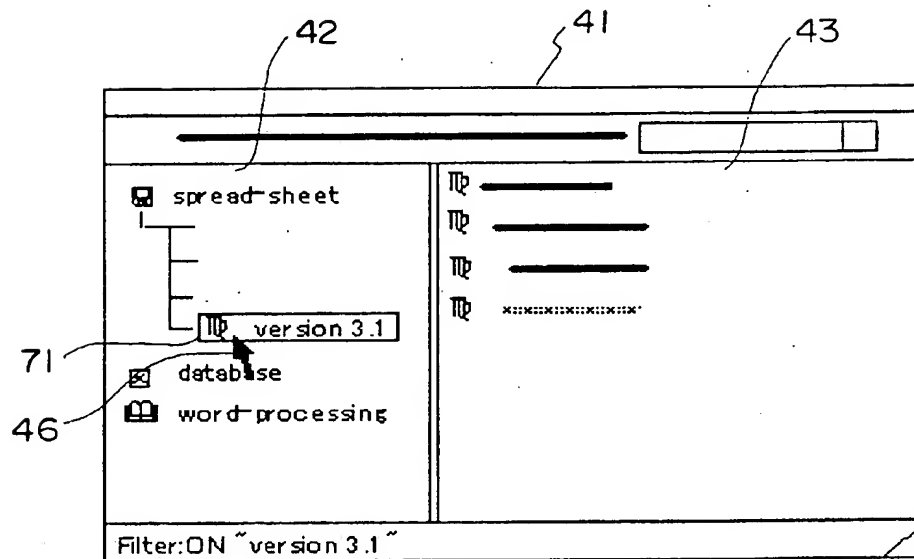
【图 1-1】

実施例 2 において、インストール領域が不足している場合のアンインストールを行う場合の機能ブロック図



【図7】

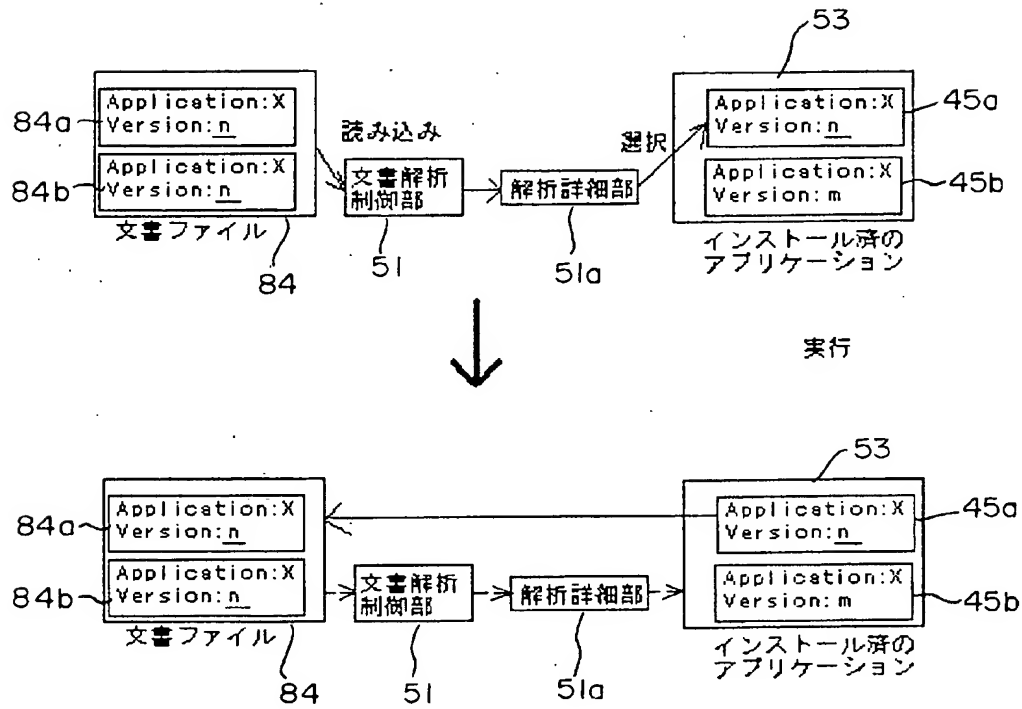
実施例2の表示装置における画面表示例(3)



また文書ファイルのリストを表示する際に、特定のバージョンのアプリケーションにより読み取ることが出来る文書の名前のみを表示させることにより、利用者がいちいちアイコンの違いを識別する必要がなくなり、さらに利便がはかられることとなる。

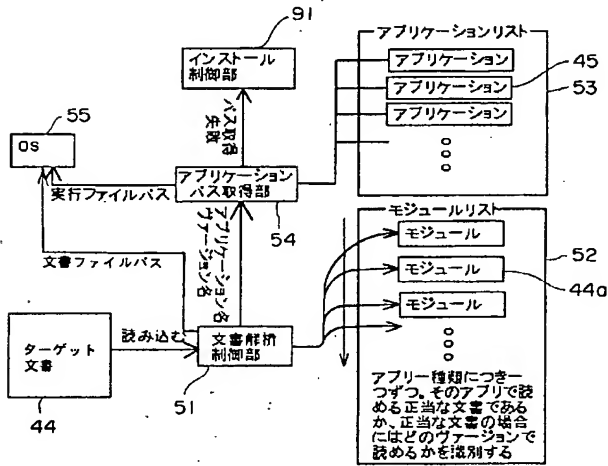
【図8】

実施例2の文書ファイルとアプリケーションプログラムとの
関連を示す機能ブロック図



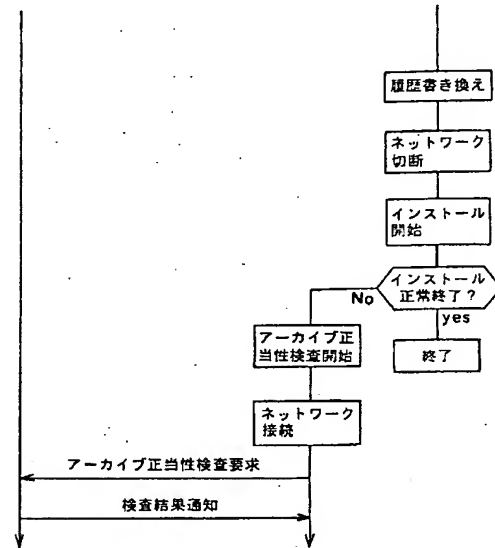
【図9】

実施例2において、文書解析制御部による解析の結果、アプリケーションリスト内に該当するバージョンのアプリケーションプログラムが存在しない場合の機能ブロック図



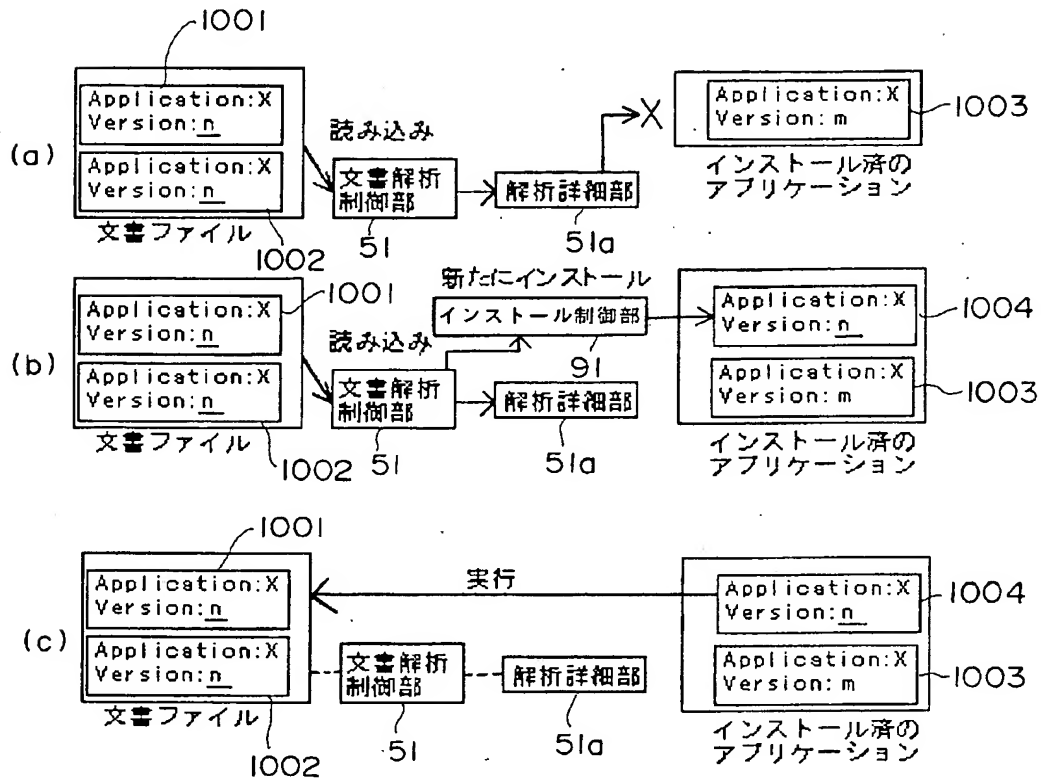
【図14】

実施例2において、ネットワークを介してアプリケーションプログラムをダウンロードする場合のシーケンス図(2)



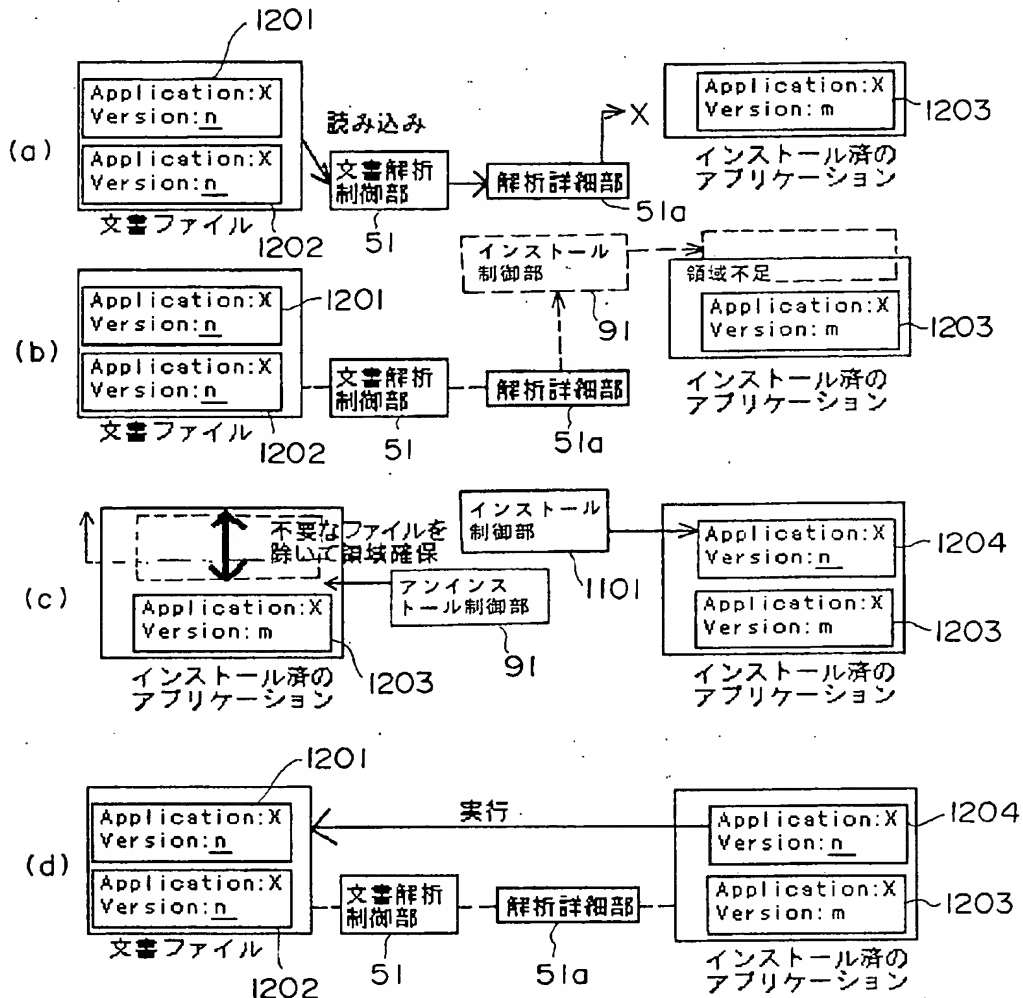
【図10】

実施例2において、文書解析結果に基づいてアプリケーションプログラムをインストールする場合の機能ブロック図



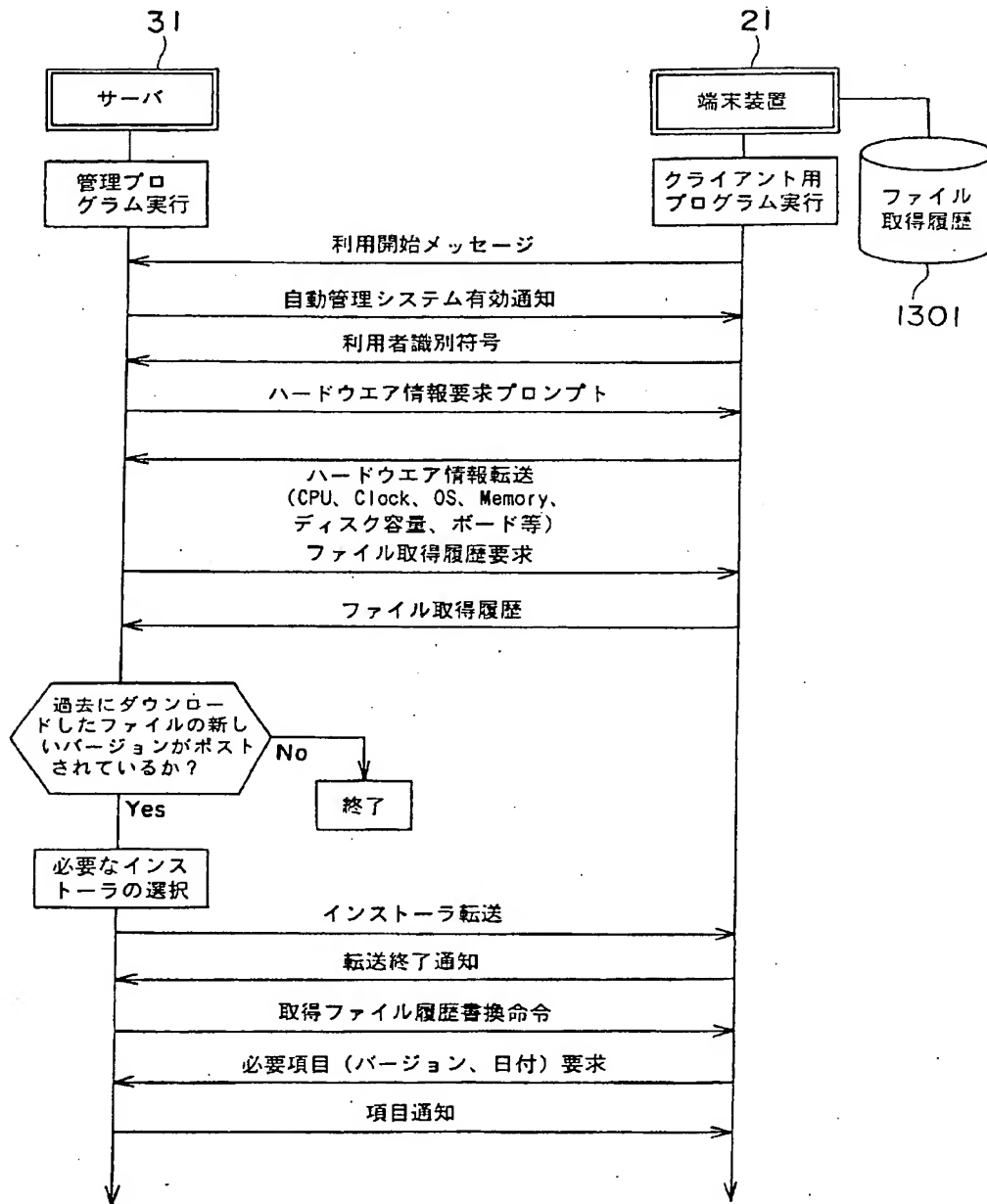
【図12】

実施例2において、アンインストールを実行して領域を確保する手順を説明する機能ブロック図



【図13】

実施例2において、ネットワークを介してアプリケーションプログラムをダウンロードする場合のシーケンス図(1)



フロントページの続き

F ターム(参考) 5B076 AC03 BB06 BB17
5E501 AA13 AC24 AC25 BA05 CB09
DA02 EA08 EA11 EB05 FA04
FA13 FA14 FA23 FB43

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ ~~FADED~~ TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ ~~LINES~~ OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.